Hartson
Hix
Editors

ADVANCES IN HUMAN-COMPUTER INTERACTION, VOLUME 3

ISBN: 0-89391-751-6

ADVANCES IN HUMAN-COMPUTER INTERACTION

VOLUME 3

H. Rex Hartson
Deborah Hix
editors

Advances in Human—Computer Interaction

Volume 3

Edited by
H. REX HARTSON
DEBORAH HIX
Virginia Polytechnic Institute and State University

ABLEX PUBLISHING CORPORATION
NORWOOD, NEW JERSEY

Printed in the United States of America

ISBN: 0-89391-751-6

ISSN 0748—8602

Ablex Publishing Corporation
355 Chestnut Street
Norwood, New Jersey 07468

Contents

Preface

When we started working in the field of human-computer interaction over ten years ago, there wasn't a field. We got strange looks from people who said "HCI isn't research; it's just common sense!" Now it is a fast-growing area of research and development. However, when we look around at advances in the field, we discover that many hard problems are still in front of us.

The theme of this series is the cooperation between behavioral scientists and computer scientists in developing human-computer interfaces. Recognition of the need for multiple roles in interface development has led to increasing diversity in this field. We welcome this as a positive indication of the contributions that a wide variety of perspectives can bring to bear on the open issues of the field. This diversity is reflected in the present volume, which includes technology, methodology, techniques, case studies, applications, user support, and tools. This is a nice continuation from Volume II, in which we stated in the preface that cooperation was needed in all the major areas of current work. Theory, modeling, methodologies, tools, and evaluation were included in Volume II; all those same topics plus technology and case studies are now addressed in this volume.

In the half-decade since Volume I of this series appeared, in 1985, we have seen a dramatic increase in the conferences, journals, and other publications that address HCI issues. The ACM SIGCHI conferences have steadily increased in size every year; 1990 saw an unprecedented attendance of more than 2,300 participants. In fact, SIGCHI itself has been the fastest growing special interest group of the ACM for several years. Also since that first volume appeared, the ACM User Interface Software and Technology — UIST — Symposium has been created. A sizable portion of the annual ACM SIGGRAPH Conference is now devoted to user interface work. Also since Volume I, the Human Factors Society Conference has increased its attention to human-computer interaction, the Interact International Conference has grown, and the HCI International Conference has begun.

One phenomenon that arises in HCI is the introduction of "gee whiz" technology, such as data gloves, touchscreens, artificial realities, and multimedia. While interfaces using this kind of technology can be glamorous and engaging in themselves, they can still be neutral with regard to usability. Technology alone can make fancy interfaces, but not necessarily good interfaces. The first chapter, by Sears, Plaisant, and Shneiderman, is an example of how behavioral analyses coupled with software skills can lead to creative ways to improve dramatically the usability of existing technology. In the case of touchscreens, technology has been limited by the imaginations of those who have applied it. This team effort has freed up the possibilities of touchscreen applications.

This relationship between the roles is also a key aspect of Chapter 2, by Browne, Summersgill, and Stradling, that highlights the important methodological differences between software development and user interface development, especially in the early phases of the life cycle. Nielsen's Chapter 3 also addresses the need for bringing the behavioral role, and especially the empiricism it offers, into the development process. Because it attempts to make an empirical process usable by computer scientists during development, this is a specific example of work that promotes cooperation among the behavioral and computer science roles in the interface development process.

Case studies are also important, especially in a relatively new field where there is a need for success stories and examples to follow. Chapter 4 by Wagner, Levinson, and Jank and Chapter 5 by Lindeman, Crabb, Bonneau, and Wehrli are both examples of case studies of interface development involving cooperating roles. Chapter 4 states the case for the role of social behavioral scientists to provide user support in specific application areas through knowledge of information needs. This chapter discusses a development project from the perspective of the interaction of a broad variety of team roles, and argues that the boundaries of HCI as a discipline must be broadened even more to encompass the specific needs of real-world application development. In Chapter 5, the application is a multimedia document retrieval system, and cooperation of team roles is stressed throughout the development process with continuous emphasis on the user's environment and support of the user in the application.

In another area of application, Chapter 6 by Glenn and Chignell addresses the usability of hypermedia, especially for browsing, again making the case for usability to catch up with new technology. Here the authors propose that the structural properties of hypermedia, so important in internal design, be used to advantage in providing visual navigation landmarks for the user. This is followed in Chapter 7 by Lee, who gives thought to the question of user support in general, especially for computer-based individual work. This chapter makes the case for user support as an important element of the interface and proposes an interaction history facility as a user support tool.

Chapter 8, by Boies, Bennett, Gould, Greene, and Wiecha, presents an outstanding example of the kind of interactive tools becoming available to support concepts such as the ones presented in earlier chapters. This chapter is especially appropriate to complete this volume because it ties together methodologies, techniques, and tools—all vitally important areas of HCI research and development.

We are pleased with this volume and the diversity it represents in this exciting field of HCI. We wish to thank the authors who have worked hard to make their chapters the best possible. We are also grateful to the reviewers who spent a great deal of time critiquing the chapters. Jo-Anne Lee Bogner and Susan Stolarski, our "wonder secretaries" over the course of this volume, have kept us in line, on schedule, and organized, helping to make tolerable the difficult job of coordinating and editing such a volume. We are especially pleased that Ablex Publishing continues to encourage us in our efforts to publish this series, in recognition of the importance of the advances that are being made in the area of human-computer interaction.

Rex Hartson
Debby Hix
Blacksburg, Virginia
November 1990

CHAPTER 1

A New Era for High Precision Touchscreens

Andrew Sears
Catherine Plaisant
Ben Shneiderman

Human-Computer Interaction Laboratory &
Department of Computer Science
University of Maryland

INTRODUCTION

One goal of human-computer interaction research is to reduce the demands on users when using the computer. This can be done by reducing the perceptual and cognitive resources required to understand the interface or by reducing the motor effort to use the interface. The introduction of alternative input devices, such as the mouse and joystick, significantly improved some user interfaces. The touchscreen combines the advantages of these other devices with a very direct method of inputting information. Users simply point at the item or action of interest, and it is selected.

While many input devices allow interfaces to be customized, increased directness distinguishes touchscreens. Touchscreens are easy to learn to use, fast, and result in low error rates when interfaces are designed carefully. Many actions which are difficult with a mouse, joystick, or keyboard are simple when using a touchscreen. Making rapid selections at widely separated locations on the screen, signing your name, and dragging the hands of a clock in a circular motion are all simple when using a touchscreen, but may be awkward using other devices. Even when a task can be accomplished with other input devices, users may have to clear their workspace for the mouse or press many keys to move the cursor.

Touchscreens have long been thought of as being simple to use. Unfortunately they have a reputation as being practical only for selecting large targets and as being error prone. Recent empirical research, as well as advances in touchscreen hardware, have dramatically improved the performance of touchscreens and the range of applications for which they can be advantageously used. Even with these advances, today most touchscreen applications emphasize the metaphor of "buttons" being pressed on the screen. Tasks such as dragging an object on the screen, moving the marker on a slider, or freehand drawing are rarely attempted with touchscreens, but we believe that touchscreens can excel in such cases. This chapter presents recent empirical research which can provide a basis for theories of touchscreen usage. We believe recent improvements warrant increased use of touchscreens. Human factors specialists, psychologists, and computer scientists have a grand opportunity to influence further developments and refine theories in these new domains.

ADVANTAGES AND PERCEIVED DISADVANTAGES OF TOUCHSCREENS

There are many advantages to touchscreens which have made them popular for public access situations.

Advantages

Directness: One of the biggest benefits of a touchscreen is its directness. Unlike indirect devices such as a mouse, joystick, or keyboard, touchscreen users simply point at the desired object, and it is selected. There is no need to remember a complex syntax, search for the input device, remove visual focus from the objects of interest, or press multiple keys to move the cursor. More importantly, there is no need for users to map hand motions to cursor motions, as required by many other input devices. Sliding, dragging, and gestural input also benefit from the touchscreen's directness.

Speed: The touchscreen is the fastest selection device for many tasks. Users do not need to reach for the input device when it is time to make a selection as they often do with a mouse or lightpen. An additional advantage in many situations is the lack of a cursor when users are not touching the screen. Users simply touch the desired location rather than touching a cursor and dragging it to the desired location.

Ease of learning: Touchscreens are easy to learn to use. Once users realize that they must simply touch the screen to interact with the computer, they quickly master simple actions such as touching buttons or dragging items across the screen. Unlike the mouse or tablet, there is no need to learn and practice spatial reorientation and hand-eye coordination (Nielsen & Lyngbaek, 1990).

Flexibility: Touchscreen interfaces offer flexibility not available with a keyboard. Each interface can be customized for each specific task performed. Users can choose which keyboard layout they prefer, QWERTY, Alphabetic, or Dvorak, since it is displayed on the screen.

No moving parts: The lack of moving parts contributes to the durability of touchscreens that has made them popular for applications such as information kiosks at amusement parks, office buildings, or museums. Unlike a mouse or keyboard, only the touchscreen must be accessible to users, making loss or damage of hardware less likely. One system, an information kiosk developed for the Smithsonian, traveled to museums across the country for two years. These touchscreens were heavily used and never failed. However, the video monitors did ultimately fail from abuse during shipping.

No additional desk space: Touchscreens free desk space for other uses. Many input devices, such as the keyboard and mouse, require desk space which may be very limited. A related benefit is that the touchscreen is in a fixed location. Unlike the mouse or lightpen there is no need to search for the device which may be hidden under papers. If the user is currently working with the computer, the screen must be accessible. This is particularly useful for applications requiring only occasional pointing.

Perceived Disadvantages

There are also some problems that have been associated with touchscreens. Many of these problems have been overcome or reduced by improvements in touchscreen technology or design strategies that have been developed for touchscreen interfaces.

Low resolution: This is one of the biggest misconceptions about touchscreens. Many people have reported on the low resolution of touchscreens. Some researchers have claimed that the resolution of a touchscreen is limited by the size of users' fingers, and others have claimed that selection of single characters would be slow if it was even possible. Recent research has shown that targets 0.4 × 0.6mm could be selected with touchscreens (Sears & Shneiderman, 1991). The same research concluded that targets 1.7 × 2.2mm could be selected as fast with a touchscreen as they could with a mouse, with similar error rates.

Arm fatigue: This could be one of the most significant problems with touchscreens. Using a touchscreen at the angle most monitors are currently mounted can lead to arm fatigue, making them difficult to use for extended periods of time. Renewed interest in reducing fatigue appears to have resulted in simple changes to the touchscreen position that will significantly reduce this problem (see the section on Workstation Design on page 24 for more details).

Parallax: When touchscreens were first introduced, the infrared technology was prevalent. Early infrared touchscreens had the touch sensing devices mounted above the surface of the monitor. When users' fingers were close enough to the screen, the infrared beams would be broken, resulting in a touch. This could occur long before the user meant to touch the screen. Newer infrared touchscreens, and all other technologies, sense touches much closer to the monitor surface, if not directly on the surface, reducing the problem with parallax. Software strategies have also been explored that reduce problems created by residual parallax by correcting for offsets created by the parallax and providing feedback to users about their exact position.

Glare and smudges: Glare and smudges on the monitor are of concern to many designers. Mounting the monitor at a better angle, using lightly ground glass surfaces, and paying careful attention to the lighting near the workstation can significantly reduce the glare problem. Smudges are unattractive and can obscure the display. Reducing smudges simply requires users to clean the monitor occasionally. On the other hand we find that some touchscreens have less problems with accumulating dust than standard monitors. In our laboratory environment, we find ourselves cleaning the mouse pad and mechanical parts more often than we clean the touchscreens.

Obscuring of the screen: The fact that users use their fingers to make a selection by touching the screen implies that the users' hand will obscure a part of the screen. Careful design of the interface, placing selectable items in locations that will keep the user's hand from obscuring the screen, can significantly reduce this problem. When possible, the handedness of users should be considered when designing interfaces, or users could be allowed to customize the software for the left or right hand.

Limited tactile feedback: Visual and audible feedback should be used to compensate for limited tactile feedback in button applications. Tactile feedback is particularly important when performing rapid button presses without watching the screen. An example is typing on a touchscreen. When users type on a traditional keyboard, the edges of the keys help orient their hands and the motion of the keys indicates when they are pressed. These cues are not available with touchscreen keyboards. Visual and audible feedback can supplement the physical contact with the screen to help compensate for the absence of key motion, but identifying when the edge of the touchscreen key is touched is more difficult. When performing tasks that involve sliding and dragging, the friction between the users' finger and the screen provides some tactile feedback. Although this problem is not unique to touchscreens, it is an important consideration when designing touchscreen interfaces. Currently research is being conducted to improve user performance for "typing" with touchscreens (Sears, 1990; Sears, Revis, Crittenden, & Shneiderman, 1991; Plaisant & Sears, 1991).

Undesired touches: When using touchscreens users may rest their hands on the screen for extra support or to reduce arm strain, or they may inadvertently touch the screen with another finger. This causes touchscreen hardware to lose track of the location users wish to touch. Research with touchscreens that recognize multiple touch locations may prove useful in eliminating this problem.

Price: Touchscreen prices are getting lower, but are still relatively expensive. Touchscreens range from approximately 350 to over 1,000 dollars. This is considerably more than most mice, joysticks, or lightpens.

Many of these problems have either been overcome or reduced, and usage is steadily increasing. Many of the problems associated with parallax and glare have been overcome by advances in touchscreen hardware. Design guidelines can significantly reduce the problems associated with obscuring the screen, the lack of tactile feedback, and undesired touches. There is renewed human factors research into reducing fatigue that appears promising. The price of touchscreens is decreasing as technology improves and touchscreen use increases. It is anticipated that when manufacturers begin producing monitors with touchscreens installed at the factory, the price of touchscreens should drop significantly.

We cannot resist mentioning some of the historical prejudices against the touchscreen. Many of the pioneer touchscreens did have severe limitations. As a result, many people still picture touchscreens as low-precision, high-error rate input devices. Touchscreens can be reliably used to select relatively small targets (approximately 2 mm square). Touchscreens do not require a large and intrusive frame glued or taped on a monitor as many early versions did. They can be mounted directly onto the surface of the monitor and all supplemental hardware can be installed inside the monitor. In summary, touchscreens have improved dramatically in recent years, and, as a result, high-precision, low-error rate tasks can now be performed using a touchscreen. Now researchers can explore new strategies and applications to guide practitioners.

COMPARISON WITH OTHER INPUT DEVICES

Touchscreens have been compared empirically to mice, lightpens, keyboards, joysticks, and other devices. The majority of human factors studies tested touchscreens against various devices for selecting predefined, stationary targets. Time and error rates were measured, and some studies measured user satisfaction. In general, these studies have shown that touchscreens are the fastest device for selecting stationary targets (Muratore, 1987; Ostroff & Shneiderman, 1988; Ahlström & Lenman, 1987; Karat, McDonald, & Anderson, 1986). Unfortunately, touchscreens have also been shown to be the most error-prone input device (Muratore, 1987; Ahlström & Lenman, 1987). However, much of this research emphasized relatively large targets, and few used alternative selection strategies that may improve user performance, making this research of limited use for higher resolution tasks, such as character selection or graphics input.

A recent study (Sears & Shneiderman, 1991) compared the touchscreen to the mouse for the selection of various size targets when using the lift-off selection strategy (the lift-off strategy will be described in the following section). This study showed that using this selection strategy can result in very low error rates for the touchscreen. It also showed that selection of very small targets (0.4×0.6mm) is possible with the touchscreen, refuting claims that the size of the user's finger determines the minimum target size. This study showed that selecting targets that are approximately the size of a character is as fast with a touchscreen as with a mouse.

Other studies have compared various input devices for selection tasks when users must also type on a keyboard. One study compared a touchscreen, mouse, and keyboard for a selection task. Results indicate that the touchscreen was preferred for tasks that do not involve a typing subtask, and the keyboard was preferred when the typing subtask was included. The touchscreen was preferred to the mouse in both situations. The touchscreen was also the fastest device for cursor positioning in both situations (Karat et al., 1986).

Unfortunately, there have been few comparisons of the touchscreen to other input devices for tasks other than target selection. Tasks such as dragging objects and outlining an object offer new opportunities for researchers. Informal observations indicate that tasks involving unconstrained movements may be significantly easier with the touchscreen than many other devices. Of course, a stylus interface or tablet may prove superior for some tasks due to their similarity to writing with a pen. Additional studies are needed to understand the range of tasks for which touchscreens can and cannot be used.

Overall, touchscreens appear to be the fastest device for selecting relatively large targets. They can also be used for selecting smaller targets if the correct selection strategies are used. Error rates can also be reduced to a point where they are negligible if the correct strategies are used. Tasks such as dragging objects on the screen or marking the border of an irregular region, also appear to be very promising with touchscreens. The last section of this chapter contains references to many papers dealing with touchscreens.

DESIGNING TOUCHSCREEN APPLICATIONS

A Model of User Interaction

We might consider a model of operation that divides touchscreen usage into seven stages. These are based on the syntactic/semantic (Shneiderman, 1987) and seven stages (Norman, 1988) models. The user:

1. Formulates a plan of what needs to be done in the task domain,
2. Examines the current computer screen to identify all touchable and nontouchable areas that represent actions and objects relevant to the task,
3. Identifies the desired touchable area by the action or object,
4. Reaches out to touch (the syntax is simply a touch) the desired area and receives feedback from hand position and from on-screen changes (a cursor, selectable areas inverting, etc.),
5. Confirms that the finger is on the desired touchable area and lifts-off to activate,

6. Confirms that the desired touchable area has been activated,
7. Interprets and evaluates the result of the touch in terms of whether the task domain goal is furthered.

This model describes interaction when the lift-off selection strategy is used with a touchscreen. Simple modifications can adapt this model for other selection strategies, or for other input devices.

The central benefit of using a pointing device rather than a keyboard is reducing the syntactic load by replacing typing with pointing. The directness of touchscreens further simplifies the task by allowing users to simply point directly at the object or action. Instead of detailed instructions about what to type (the syntax) to select an action or object, users simply touch a visual representation of the object or action (the semantic description). Touchscreens avoid the distraction of looking from the screen to the input device and back while remembering the desired syntax. If designers choose a proper set of touchable objects and actions, then progress can be rapid with low error rates.

The traditional touchscreen strategy has been *land-on*, in which the first location that users' touch initiated an action. This is acceptable, even preferable, when there are a few large targets. However, as the task domain complexity increases, the number of choices can increase dramatically and undermine the efficacy of the land-on strategy.

With high resolution touchscreens that support continuous feedback, the finger touch may produce a cursor that can be dragged across the screen, and activation occurs when the finger is lifted off the surface. This can not only improve the selection of menu items or buttons but also open a new world of interaction techniques such as the direct manipulation of metaphors (moving a cursor on a slider, selecting a color on a color wheel, etc.), freehand drawing, and symbolic gestural input. There are many other possible strategies and combinations of strategies that can be used.

The following sections will discuss various interaction techniques, and how each technique can be used with a touchscreen.

Traditional Button or Menu Selection

Menu or button selection tasks typically require the selection of predefined targets represented on the screen. The majority of current applications for touchscreens involve tasks such as these. This section discusses factors that play important roles in button selection tasks including the visual representation, size, and location of targets and several alternative selection strategies.

Visual representation of touchable areas. Users are constantly confronted with the question of what is a touchable area. A consistent principle or small set of principles greatly reduces the burden on users (Apple, 1987). Possibilities include: realistic button shapes, rounded rectangles, shadowed boxes, distinctive color text, distinctive color background, tabs on a book, or standard icons. Designers must remember that it can be frustrating for users who identify what they think is a selectable object, try to select it, and discover that it is not selectable after all. If instructions or icons are used, they should be chosen to be consistent with the tasks and users' expectations.

There is no simple solution for indicating what is selectable. In some systems, all selectable objects appear in the same shape (Figure 2) or color (Figure 1). Once users learn this rule, all targets can easily be identified. Other systems place simple instructions on the screen, possibly near each different target. The instructions may be as simple as "Touch the desired amount," indicating what is selectable and how to select it (Figure 3). The goal is to choose an option that works best for the tasks being performed, and to be consistent once the choice has been made.

Making objects significantly different from the remainder of the screen can result in users ignoring the remainder of the screen. In some

GOVA - Introduction PAGE 1 OF 2

Welcome to **GOVA,** the Guide to Opportunities in Volunteer Archaeology. **GOVA** will tell you about **archaeological projects** all over the world **(see Map 9)** that welcomes **volunteers**. You can use **GOVA** to answer **practical questions** like **how to join a dig,** or how much it **costs**.

To Use **GOVA,** simply touch a blue word with your fingertip-- not your fingernail-- and release. For example, if you touch the blue word "volunteers" at the third line of this page, you will get an article about volunteers.

To turn to the next page touch the blue word NEXT PAGE.

INDEX **NEXT PAGE**

RETURN TO MAP 9, THE WORLD **RESTART**

Touch a highlighted word with your fingertip

Figure 1. Hyperties, a hypertext system, allows users to traverse a database of articles including text, graphic and videodisc images. All selectable items are in bold (these appear in light blue on the actual system).

situations, designers may not want to make it obvious what is selectable: when the designer wants to force users to explore the entire screen, when almost everything is selectable, or when an existing image cannot be overlaid or altered. In these situations, the designer can either force users to select what they think is selectable, or a mechanism can be provided to reveal all selectable regions. A special key or a "Reveal" button may make all targets temporarily visible. Targets may also be shown when users try to touch a nonselectable region. This method can be both frustrating and slow if there are many selectable regions on the screen, making a "Reveal" button preferable when possible.

As a general guideline, if users have specific goals, making targets obvious may speed performance. However, if the purpose is to explore and gain general knowledge about the system, it may be desirable to make the targets blend in with the remainder of the screen.

Feedback. Feedback plays an important role in every user interface. Feedback indicates where the user is currently touching the screen, that an action could be taken, or that an action has been taken. Feedback provides confirmation to users that the correct actions are about to be or have been performed, as described in the model presented earlier.

If users are allowed to drag their fingers before making a selection it is often advantageous to provide a cursor near the users' fingers showing exactly where a selection will be made if they lift their fingers. This is particularly important if the targets are very small, less important if targets are large. It is also important to indicate when a selection is possible. If the lift-off strategy (or any other strategy that uses the removal of the users' fingers as input) is used, then as users drag their fingers onto a target, users should receive feedback that a selection could be made. This could be visual, by inverting or flashing the target, or audible, by making a short tone. Once a selection has been made, feedback should indicate that an action is about to be taken. This confirmation could be visual or audible. Visual confirmation has the advantage that the specific action to be taken could be indicated (the selected target could be inverted temporarily indicating exactly what action will be taken). Audible feedback has the advantage that users eyes can be off the targets, however, audible tones are more difficult to distinguish (making them less useful for indicating the exact action to be taken). If audible feedback is to be used, the volume must be set carefully. If the system is used in a public place or in an open work area, loud tones may be very annoying; if the tones are too soft, they may be missed. User control is useful in this situation.

Feedback about possible selections can often replace the cursor. In the example in Figure 4, as users drag their fingers across the calendar, they select a day by lifting their fingers when the desired day is highlighted. Highlighting days acts not only as a cursor, but also as an indication of possible selections.

Target size. Choosing the appropriate size for touchable regions is important when designing the user interface. The size of targets can depend on the physical size of the display, the number of targets to be presented, the relation among targets, and the type of application. The physical display size limits the target size. The number of targets to be presented and the relation among these targets may influence the target size. If users will be selecting one or more closely related targets, then they should be presented simultaneously. This allows users to view all options at one time, and to make an informed decision about which ones to select. The type of application may also influence target size. If making an error may result in serious problems, targets should be made large enough to assure that they are accurately selected. Alternative selection strategies that require confirmation may also be used if errors are critical. Other possibilities include changing the size of the touchable regions associated with a target. If targets are not close together, it is possible to make the touchable region larger than (and possibly offset from) the visible target. This allows users to make accurate selections even if they touch slightly off the visible target.

Fitts' Law provides some insight when choosing target sizes. Fitts' Law relates selection times to the distance that must be moved and the width of the target (Fitts, 1954). The general form of the law is $T = a + b[\log(2d/w)]$, where a and b are constants, d is the distance that must be moved, and w is the width of the target. This law has been demonstrated with various input devices, however recently a slight modification has been suggested for use with touchscreens (Sears & Shneiderman, 1991). This modification suggests using Fitts' Law twice, once for when users move their fingers to the screen, and a second time when they move their fingers on the screen. In general, Fitts' Law provides a method of estimating the time it will take users to select targets. It allows designers to predict how increasing the target size will affect the time to select that target. It also provides insight into how the spacing between targets may relate to selection times (if multiple selections will be made from a single screen).

Layout of targets. The layout of targets refers to the location of each target on the screen and their location relative to each other. Targets should be placed consistently on the screen. Buttons for Quit, Back, Next, Cancel, etc., should be placed consistently from one screen to the next. Careful placement of targets on the screen can reduce or eliminate certain problems. Placing targets low on the screen reduces the amount of the screen obscured by users' hands when they are making selections. Placing targets near the edge of the screen has both advantages and disadvantages. When targets are near the edge of the screen, the edge acts as a barrier making it impossible for users to miss the target by touching too far to one side. However, placing targets near the edge of the screen may make selection difficult depending on how far the touchscreen is recessed into the monitor. Of course, the handedness of users should also be considered. Although it may be difficult, if not impossible, to determine the handedness of potential users when designing the interface, it may be possible to design the interface so neither right- nor left-handed users suffer.

The spacing between targets can play an important role in the speed and accuracy of selections. Using information from Fitts' Law, we can understand the basic relationship between target spacing and selection rates. As targets are moved closer together, selections are faster and error rates may increase. As targets are spaced farther apart, selections will be slower, but error rates will decrease. The combination of spacing and target size plays an important role in determining both selection times and error rates. If the application requires fast, accurate selections with minimal attention, targets should not only be larger (as discussed earlier) but spaced farther apart to reduce the likelihood of incorrect selections. Target size and spacing also play an important role in deciding which selection strategy to use (see the section on Selection Strategies).

Response time. System response time must be quick, but not too quick. Systems must respond fast enough so the user knows that the computer received the input, either by immediately showing the result of the action or giving some feedback acknowledging the input was received. For example, when users select a button, the system should immediately indicate that a selection has been made, even if processing the input will take a few seconds. If the system is too slow indicating that a selection has been made, a user may attempt the selection a second time, resulting in unwanted input.

It is possible for a system to respond too quickly. If the changes to the screen are very subtle, users may not notice the update if it occurs too quickly. In these situations, it is sometimes desirable to either provide additional feedback or slow the response slightly so users can observe the changes occurring.

Selection strategies. A summary of when each strategy may be appropriate is provided at the end of this section.

• *Land-on:* When touchscreens were first introduced, hardware limitations resulted in a single selection strategy: land-on. Since only the location of the initial screen touch was available, the land-on strategy resulted in the selection of the target that was at this location (Murphy, 1987; Potter, Weldon, & Shneiderman, 1988; Potter, Berman, & Shneiderman, 1989). Using the land-on strategy, users touch the screen and the location of the touch is compared to the locations of the targets on the screen. If the touch is on a target, that target is selected. Otherwise, users must remove their fingers from the screen and make another selection attempt. Land-on can be used when targets are large enough to assure that users will not inadvertently touch an incorrect target. Targets approximately 2.0–2.5cm per side can be accurately selected with one attempt using the land-on strategy, depending on the touchscreen technology (Sears, 1990; Beringer, 1989; Hall, Cunningham, Roache, & Cox, 1988).

There are some applications where the targets must be selected accurately with minimal attention. Applications such as automobile or helicopter controls require that the selections be made while diverting users eyes for a minimal amount of time. When using the land-on strategy users simply look at the screen one time and touch the desired target. Since selections are made when users first touch the screen, additional finger movements are unimportant. The land-on strategy may be preferred for applications that require selections with minimal attention as long as targets are sufficiently large. It is important when using the land-on strategy to provide feedback indicating a selection has been made. The feedback can be as simple as temporarily inverting the target, or making an audible tone. If feedback is not provided, users may make a second selection before realizing that the first attempt was successful. As touchscreen hardware advanced, allowing continuous feedback about the location of the touch, new, more advanced selection strategies were developed.

• *First-Contact:* The first-contact strategy uses the continuous feedback about the touch location to select the first target the users' fingers touch. If users touch a target when they first land on the screen, it is selected. If they miss all targets when they first touch the screen (land on a nonselectable part of the screen), they can drag their finger to the desired target. As soon as users touch any target, it is selected. Once a selection has been made users must remove their fingers from the screen before another selection can be made (Potter et al., 1988, 1989; Murphy, 1987).

The first-contact strategy is useful when targets are either relatively large, or smaller and placed sparsely on the screen. If targets are relatively large, the first-contact strategy will function the same as land-on. If targets are smaller but placed sparsely on the screen, users can attempt to touch the desired target, and if they miss, they can drag their finger over the target until it is selected. A cursor can be placed near the users' fingers indicating exactly where a selection will be made when using the first-contact strategy. First-contact is not recommended for densely packed, small targets. Once again it is very important to provide feedback informing users when a selection has been made.

• *Lift-off:* Lift-off also uses continuous feedback about the touch location. When using this strategy, the selection is made when users' fingers are removed from the screen. Users touch the screen, they can then drag their fingers to a new location if desired, and lift their fingers to make a selection (Potter et al., 1988, 1989; Murphy, 1987; Weisner, 1988). A cursor can also be used with the lift-off strategy. The cursor may be placed above the users' fingers if small targets are to be selected, indicating exactly where the selection will be made. This strategy allows densely packed targets to be selected with minimal errors.

The lift-off strategy provides additional accuracy and user control at the cost of additional perceptual and cognitive effort. It is useful when targets are relatively small or densely packed, but can be used for any size targets. Our experience shows that this strategy, combined with stabilization software, allows the selection of 0.4 × 0.6mm targets (Sears & Shneiderman, 1991). Stabilization takes the raw data from the touchscreen, and converts it to a specific screen coordinate. When users touch the screen without moving their fingers, stabilization software allows a single screen coordinate to be selected. This is not possible with many touchscreens if stabilization is not used (see the section on Filtering Raw Touchscreen Data for a discussion of stabilization).

The lift-off strategy is useful in many applications and is recommended for small or densely packed targets, although it can be used for any set of targets. Simple menus, including pull-down menus, can be traversed using the lift-off strategy. The lift-off strategy has also been used successfully to select days from a calendar (Figure 4 and Figure 8) or a room from a floor plan (Figure 5). Users often use interfaces as if the land-on strategy were being used, and use dragging when errors occur or targets are small. Novice users rapidly learn how to drag objects, or a cursor, until it is placed where they want it.

• *Land-on/Lift-off:* In most situations either land-on or lift-off are adequate, but occasionally more complex strategies are needed. The most common reason for using more complex strategies is to provide additional confirmation before selections are made. Additional confirmation may be needed if the action is difficult to reverse, such as deleting files, or in life-critical situations. For these situations, the land-on/lift-off strategy may be used (Weisner, 1988). This strategy not only requires that users *land-on* the target, but that they also *lift-off* the same target (sometimes with the additional condition that the user never leave the target). This is similar to a mechanical button with a protective barrier around it to prevent accidental activation. The additional confirmation this strategy provides is sufficient for many applications where land-on or lift-off alone are not adequate.

• *Sequential-touch strategies:* When additional confirmation is needed and the land-on/lift-off strategy is not sufficient, a strategy that uses a sequence of touches may be what is needed. These strategies require a sequence of touches, possibly simply selecting the desired target followed by a confirmation button, before an action is taken (Murphy, 1987). The confirmation button may be a special button or the same button the user has just selected (selecting the button a second time acts as confirmation). The sequence of selections can use any of the previously mentioned strategies for each individual selection. Sequential-touch strategies are useful for applications when accuracy is critical but speed is not. Many sequential touch strategies have been devised.

• *Summary:* We close this section with a small sample of some possible selection strategies and a brief summary of recommendations.

Land-on: Relatively large targets, 2.0-2.5 centimeters per side or larger.
First-contact: Relatively large targets. Smaller targets when placed sparsely on the screen.
Lift-off: Any size targets (especially when densely packed).
Land-on/Lift-off: Simple confirmation needed. Other tasks are being performed that may result in false selections if other strategies are being used.
Sequential-touch: Confirmation needed.

A few examples.

• *Touchscreen keyboards:* There are many factors that go into designing a touchscreen keyboard including the size, spacing, and layout of the keys. Research indicates that for typing text the QWERTY keyboard is preferred by typists and does not slow nontypists significantly (Norman & Fisher, 1982). However, when using a touchscreen keyboard, it is possible to allow users to select the keyboard style they prefer, QWERTY, Alphabetic, Dvorak, or any other known layout. After choosing a layout, the size and spacing of the keys must be chosen. This may be influenced by the size of the display area and the selection strategy to be used. If the desired result is to have relatively fast typing, the designer may choose to use the land-on strategy with larger keys. However, larger keys require additional display area, and will require more time for users to move from one key to the next. If the keyboard must fit into a small area, then smaller, closely packed keys may be used with the lift-off strategy. An evaluation of a touchscreen keyboard, using the QWERTY layout with keys 2.27cm per side, indicates that users can type approximately 25 words per minute (compared to typing speeds of approximately 58 wpm on a standard keyboard for the same users) (Sears, 1990).

• *Conference message system:* A conference message system developed by Cognetics Corporation used a touch only interface, combined with a video camera, paper, and a pen, for sending and receiving messages from other conference attendees. This system had very realistic looking buttons and high-resolution graphics. The touchscreen was used to select options as well as to move through a directory of attendees. The lift-off selection strategy was used allowing easy selection of actions with few errors.

• *Cash register:* A touchscreen cash register has been developed by Touch Industries and is in use in high-volume commercial installations such as Ben and Jerry's Ice Cream stores and the Library of Congress gift shop (Figure 6). This system allows cashiers to enter either a product number or use a menu system to select the product; touch keypads are presented for entering dollar amounts. Color coding is used to group keys by product type and function.

• *Information kiosk/product ordering system:* Many information kiosks have been developed taking advantage of touchscreens. One example (developed by Carroll Touch), for the Florsheim Shoe Stores, not only allows users to view video disk images of shoes but also allows them to order shoes. Audio recordings are played back to users as they traverse the database of shoes. For applications like this one, with large buttons, any touchscreen hardware and selection strategy can be used.

• *Process control/automated manufacturing:* Touchscreens have also been used successfully in process control applications. Figure 7 shows a complex control station, Honeywell TDC3000, that uses touchscreens.

• *Sports statistics system:* One company has developed a product, using a touchscreen, that allows easy collection of statistics at basketball games. The number of each player is on the screen, with a representation of the court. When a player takes a shot, the statistician simply touches the player's number, the location from where he took the shot, and whether it was good or not. At the end of the game complete statistics are easily computed.

Direct Manipulation of Metaphors

The directness of the touchscreen as an input device makes it desirable for direct manipulation interfaces. Not only are users directly manipulating the computer environment, they are doing it by simply touching the desired object with their fingers. Touchscreens have been used for manipulating the hands of clocks (Figure 8), moving markers on time lines (Figure 4), painting (Figure 9), and moving the needle of a compass. Several products allow the mouse to be replaced or complemented by a touchscreen.

Direct manipulation provides additional methods to get input from the user. Any task that could be performed by simple button selection, can be thought of as directly manipulating the button. However, traditional button selection used very little of the information from the touchscreen as actual input. Only the location of the initial touch and the lift-off were actually used as input (intermediate locations were simply used to guide the user). With direct manipulation input, more information is used. Every location users touch is potential input and may affect the display as they manipulate it. With the use of this additional information comes the cost of more complex strategies to take advantage of it. In general, only the current and previous locations that users touch are important since the interface is updated continuously. For instance, when dragging the needle of a compass, the current location is needed to draw the new needle, and the previous location allows the old needle to be removed. For this reason, the system must continuously track the users' fingers, while recording the location of the previous touch.

Many tasks can be performed using direct manipulation and touchscreens:

- *Rotating objects:* Rotating the hands of a clock or the needle on a compass are two examples that have been implemented (Figure 8). Users simply touch the hand and drag it to the new location. Some applications allow users to simply touch the new location, and the hand (if there is only one) jumps to the new location (updating the corresponding value). Other possibilities include rotating knobs. Users could simply touch near the knob (grabbing the part of the knob closest to their fingers) and move their fingers around the knob until the desired value is reached. Once users have control of the knob, rotational movements can be made very accurate by increasing the distance between the users' fingers and the center of the object being rotated.
- *Linear movements:* There are many tasks that can be performed by making linear movements. A practical example is a scheduling system with a linear time line. The time line not only allows fast, accurate input, but also indicates the duration of the event specified. Once the markers have been placed on the time line, they act as simple sliders (Figure 4). Another example is a graphic query interface that uses sliders to indicate dollar amounts and time periods and button selec-

tion to select the countries of interest (the buttons are shaped like the countries) (Figure 3).

Gestures and Freehand Drawing

Freehand input and gestures allow many novel applications. Freehand input allows users to move their hands across the screen with every location they touch being used as input. Gestures allow users to draw shapes or letters which are used as input. Every location they touch is used as part of the input and helps determine the final action that will be taken. Freehand input includes tasks like drawing in a graphics package, marking the border of a region that is of interest, or signing your name. In some situations, the locations are analyzed to determine the pattern, gesture, or letter users made. In other situations the entire history of where the user touched is saved and used as input (as in the case of signing your name). There are applications, like signing your name, that are easier when using a stylus to touch the touchscreen. But many of these applications are significantly easier with a touchscreen than they are with a mouse, joystick, or keyboard.

Freehand input offers a more powerful method of input than either target selection or direct manipulation. Freehand input can be used to perform any target selection or direct manipulation task. In addition, many tasks that cannot be accomplished easily with either of the other methods can also be performed. Instead of simply saving the previous touch location, freehand input requires that a longer history be maintained, possibly the entire history from the time the users first touch the screen until they lift their fingers. In some cases, the time when each touch occurred may also be needed.

A simple example of gestural input is switching something on or off by rapidly moving fingers up or down on the screen, followed by lifting the finger from the screen (Figures 10a, 10b). This simulates the motion used to operate many light switches and should be easy for users to understand and use. Another example is a U-shaped switch. Users touch one end of the switch, and must drag their fingers through the U-shaped path to toggle the switch (Figure 10c). This could be used in situations where precautions must be taken to prevent errors. A third example is a text editor. Users could touch the letters, words, or paragraphs of interest, followed by a D-shaped gesture to delete it (Figure 10d). Other gestures could be used for copying, moving, or modifying text.

Storing an entire history of touch locations and times can consume a great deal of storage space. There are many methods for reducing the amount of storage required to maintain the history. Data compression techniques can be used, but may slow the processing of data too much.

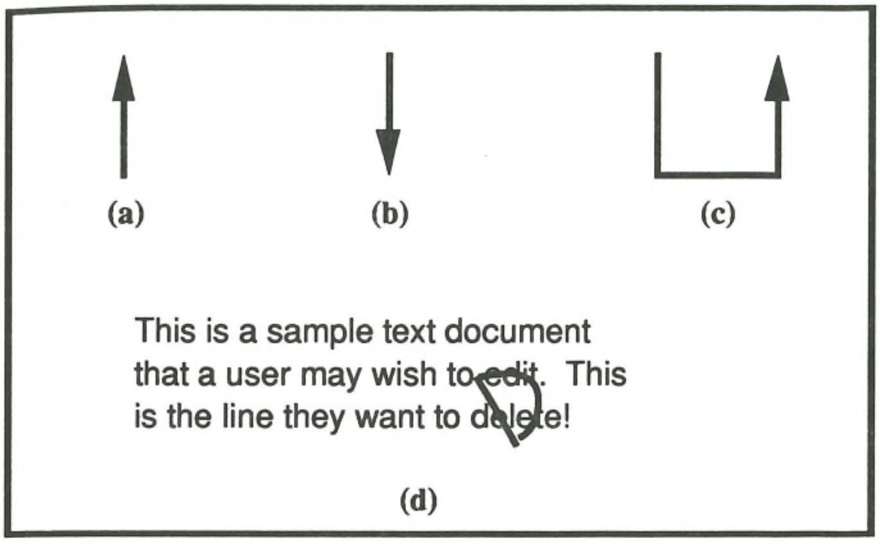

Figure 10. (a) A simple gesture to turn something on. (b) A simple gesture to turn something off. (c) A more complex switch that requires a U-shaped motion. (d) A D-shaped gestures deletes the selected text.

Other methods reduce the amount of history that is actually maintained. Sequential inputs often indicate very little or no movement of the touch location. It is possible that movements must be larger than some minimum before they are stored. Although this method can reduce the resolution of the touchscreen, the minimum can be set to eliminate this problem. This method eliminates a great deal of unnecessary data if users are not moving their fingers. Another alternative is to periodically save the touch location. This method also reduces the resolution, but in a different way. If the users make very fast movements, it is possible that important intermediate values may be lost.

Combining Button Selection, Direct Manipulation, and Freehand Input

There are many situations when using a single input method is not sufficient. An application may have sliders to indicate dollar amounts and buttons to select an item. This application could use both button selection and direct manipulation. In addition, users may be allowed to circle a set of buttons they wish to select, requiring a freehand input. Many interfaces may be easier to use if multiple input methods are used. It is important to provide input methods that appear natural for the tasks, limit errors, and work smoothly together. For instance, if sliders are being used near buttons, the first-contact strategy should not be used to select the buttons but may be used to move the slider. If freehand input is allowed anywhere on the screen, it is likely that some strategy that provides confirmation should be used to select buttons (in this case first contact is clearly not appropriate). It is important that when users are inputting values that they do not accidentally select a button or move a slider. Careful attention must be paid when designing an interface with multiple input techniques to ensure that they blend together smoothly.

TOUCHSCREEN HARDWARE AND FILTERING TOUCHSCREEN DATA

This section summarizes touchscreen technologies and discusses filtering the raw data provided by the touchscreen.

Hardware

Touchscreen hardware has improved dramatically in recent years. Touchscreens have resolutions as high as 4096 × 4096 for a standard IBM PC monitor, parallax problems have been significantly reduced, and continuous feedback of the touch location has become common. There are five main touchscreen technologies: membrane, capacitive, surface acoustic wave, infrared, and piezoelectric. One promising technology due to a high resolution, low parallax, and high percentage of light transmitted through the touchscreen is a laser-scanned touchscreen (Garwin & Levine, 1989). This technology is not commonly available and is still considerably more expensive. The remainder of this section discusses the five major technologies briefly, pointing out benefits or liabilities. For each technology, the amount of contact necessary to activate the touchscreen, the resolution of the technology, and the amount of light that is transmitted through the touchscreen hardware will be discussed.

- The amount of contact necessary to activate the touchscreen is important in many situations, including tasks that involve dragging an object on the screen. Pressing hard against the screen while carefully dragging an object can be difficult. On the other hand, having the screen activate before the user actually touches it, as can happen with infrared touchscreens, can be disconcerting.
- The resolution, if too low, can limit the tasks for which the touchscreen can be used. If the designer plans only to use large targets, the resolution of the touchscreen is less critical. When small targets will be

used, stabilization software and selection strategies must complement a higher resolution touchscreen.

- Finally, the amount of light transmitted through the touchscreen hardware is important when viewing the screen. The more light blocked or reflected by the touchscreen hardware, the more the display will change in appearance. Colors can become distorted, and reading may become difficult. For example, on a high-quality graphic workstation, it would be important that the image quality not be lowered by the touchscreen.

Carroll (1986) and Carroll Touch (1989) provide the most complete reviews of touchscreen hardware. Stone (1987), Pickering (1986), and Sherr (1988) also reviewed touchscreen hardware.

- *Membrane Touchscreens:* Membrane touchscreens typically consist of two thin layers of material separated by clear separator dots placed over the monitor. When users apply a small amount of pressure to the screen, the two layers of material make contact, indicating where the touch is occurring. The resolution of membrane touchscreens does not depend on the physical size of the screen, providing up to 4096 × 4096 touch points on a PC monitor. The percentage of the light transmitted through the touchscreen material is the lowest of all technologies, averaging between 70 and 80%. Membrane touchscreens can be operated by a finger, gloved finger, or any stylus (Carroll, 1986; Carroll Touch, 1989).
- *Capacitive Touchscreens:* Capacitive touchscreens consist of a single layer of material placed over the monitor. When users touch the monitor with a finger (or other conducting stylus), they cause a change in capacitance which is sensed by the touchscreen. Once again, the resolution of capacitive touchscreens does not depend on the physical size of the screen, providing up to 1024 × 1024 touch points on a PC monitor. Capacitive screens require less pressure than membrane screens to be activated, making them better for tasks that involve dragging objects on the screen. This technology tends to respond slower to users' touches, which may lead to problems with rapid selections. The amount of light transmitted through a capacitive touchscreen is higher than that of membrane touchscreens, averaging about 85% (Carroll, 1986; Carroll Touch, 1989; Stone, 1987).
- *Surface Acoustic Wave Touchscreens:* Surface Acoustic Wave (SAW) touchscreens also consist of a single layer of material over the monitor. When users place a finger on the monitor, the water content of their finger absorbs energy allowing the touchscreen to sense the touch location. Currently the resolution of SAW touchscreens is limited to 0.8mm per touch location. This is poorer than either capacitive or membrane systems. Resolution of at least 6.4mm per touch location

proves to be adequate for most tasks making the resolution of SAW touchscreens high enough for almost any task. SAW touchscreens require that the users' fingers or another soft stylus activate the screen. SAW touchscreens typically require more contact with the screen than capacitive but less than membrane systems, in order to activate the screen. This technology has the ability to provide a limited third dimension (Z axis) and a higher percentage of light being transmitted—92% on average. Although the third dimension exists (determined by amount of contact by the users' fingers), it is usually limited to three or four useful values at the present time (Carroll, 1986; Carroll Touch, 1989). Higher resolution SAW touchscreens are expected in the near future.

- *Infrared Touchscreens:* Infrared touchscreens do not require additional material to be mounted on the monitor surface; instead a frame is placed around the monitor. The frame contains infrared transmitters and receivers that identify when users place their fingers on the screen. Any object that blocks the path of the infrared light from one side of the monitor to the other activates the touchscreen, including fingers, pens, or any other stylus. Infrared systems require little or no actual contact with the screen, which can lead to increased problems with parallax and precise detection of when and where a user's finger lands on or lifts off the screen. The resolution of infrared touchscreens can be as fine as 3.2 mm per touch location, the second lowest of the five technologies, and is high enough for most tasks. One-hundred percent of the light from the monitor is transmitted through the touchscreen (Carroll, 1986; Carroll Touch, 1989; Stone, 1987).
- *Piezoelectric Touchscreens:* Piezoelectric touchscreens have a single pane of glass placed over the monitor. Four pressure sensing crystals are placed under the corners of this pane of glass that measure the pressure from user's fingers. When users touch the screen, the system uses the pressure at each corner to calculate the location of the touch. Piezoelectric touchscreens provide the lowest resolution at approximately 60×60 touch locations on a PC monitor. Piezoelectric touchscreens require more pressure than all other technologies before they are activated and may fail to recognize a touch if the users' fingers are pressed against the screen too slowly. Approximately 92 percent of the light from the monitor is transmitted through the touchscreen hardware (Carroll Touch, 1989).

Summary. All of the technologies can be used in a variety of applications. Research into providing higher resolution touchscreens that allow a higher percentage of light to be transmitted through the touchscreen continues. Although at one time it was common for touchscreens to provide only the location of the initial touch, current touchscreens typically provide continuous feedback, indicating exactly

Table 1. Some Characteristics of the Five major Touchscreen Technologies.

Technology	Maximum resolution	Transmitted light	Pressure needed (1 = least)
Membrane	4096 × 4096	70–80%	3
Capacitive	1024 × 1024	85%	1
SAW	0.8mm physical limit	92%	2
Infrared	3.2mm physical limit	100%	None
Piezoelectric	60 × 60	92%	4

where the users' fingers are touching at all times. The ability to detect multiple touches at once has been explored and should provide interesting research and applications if it becomes commercially available. The resolution of all of these technologies is sufficient for most tasks. However, when tasks involve the selection of small targets, or fine manipulation of objects, infrared and piezoelectric touchscreens may not provide sufficient resolution. The following is a table that contains a brief summary of some important aspects of each technology.

A partial list of touchscreen manufacturers appears in Appendix A. Many manufacturers will install touchscreens on monitors customers supply.

Filtering Raw Touchscreen Data

Touchscreens typically return three values: the touch status, a X coordinate, and a Y coordinate. The touch status indicates whether users are their fingers from the screen, and when they first touch the screen. The touching the screen or not. This value indicates when users have lifted X and Y coordinates indicate where the touch is located in the touchscreen coordinate system.

Many of the earliest touchscreens returned only the location of the initial touch and no status. As technology advanced, it became standard to return a continuous stream of data indicating exactly where the users' fingers were touching at all times. This led to many new strategies which in turn led to the ability to perform many new tasks with touchscreens.

Touchscreens may operate in a coordinate system that is different than that of the display, requiring the touch location to be converted to display coordinates. This conversion is a simple linear transformation. Some touchscreen manufacturers provide software that, given the display coordinates, automatically converts the touch location for application programmers.

Continuous feedback of the touch location can also lead to several problems: instability, false lift-offs, and false land-ons. Some touchscreen manufacturers have solved these problems in the software provided with the touchscreen; others leave the solution to the programmers.

- *Instability* is the inability of the touchscreen hardware to return a single display coordinate when users hold their fingers still. Instability has been dramatically reduced by touchscreen manufacturers. If it is necessary to provide additional stability, the directness of the touchscreen should be preserved. The cursor on the stabilized touchscreen should follow users' fingers accurately. Some algorithms cause the touch location to lag behind users' fingers if they move too rapidly reducing the sense of direct control users expect from touchscreens.

One study explored several strategies for stabilizing the touch locations. One algorithm was chosen that preserved the directness of the touchscreen and comparisons were performed with the nonstabilized touchscreen and a mouse (Sears & Shneiderman, 1991). Stabilization resulted in faster, more accurate selections of small targets while preserving speed and accuracy for selecting larger targets. Stabilizing the touchscreen also resulted in significantly higher preference ratings for the touchscreen.

- *False lift-offs* are the inadvertent removal of fingers from the touchscreen. This can occur when using the touchscreen to drag objects across the screen, leading to a frustrating experience for users.
- *False land-ons* are a similar problem occurring when the touchscreen is too sensitive and indicates that users have touched the screen when they did not mean to.

The problem of false lift-offs and land-ons has also been addressed by touchscreen manufacturers. It is common for the software provided with the touchscreen to include several parameters that can adjust the minimum time users must touch the screen before it is recognized as a land-on, and the minimum time they must lift their finger before the lift-off is recognized. If these parameters are not provided, programmers can incorporate them in the application software. When using the lift-off strategy it is also useful to filter the last touch locations to account for the sliding that may occur when users remove their fingers from the screen.

WORKSTATION DESIGN

Workstation design with touchscreens is important since the additional layers of material used on many touchscreens can result in optical interference and glare. Often users move their screens to reduce glare, but if the touchscreen is to be mounted at an angle to reduce arm fatigue, moving the screen may be impractical. Lighting in the area of a touchscreen workstation should be designed to reduce glare.

Figure 2. This conference mail system allows users to easily send and receive electronically scanned handwritten messages. All selectable items are represented by realistic three-dimensional buttons. (Copyright © 1989, 1990 by Cognetics Corporation. Reprinted by permission. All rights reserved.)

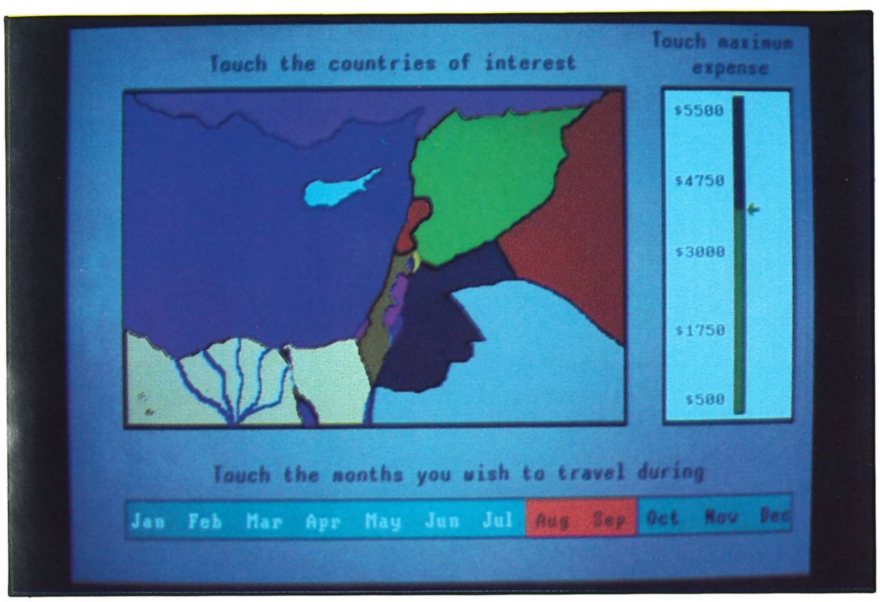

Figure 3. This is an example of a graphic query interface. Time period and cost are entered using sliders, and locations can be selected by touching the countries of interest.

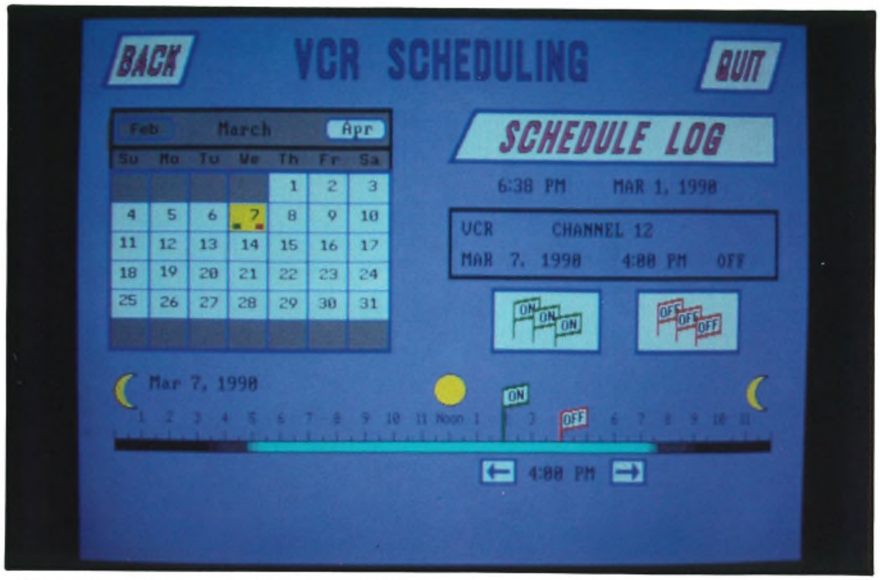

Figure 4. This scheduler uses a 24-hour linear representation of a day. Users drag an ON flag to the desired position on the time line to turn a device on; then an OFF flag to set the off time. (Copyright © 1988, 1990 Custom Command System. Reprinted by permission. All rights reserved.)

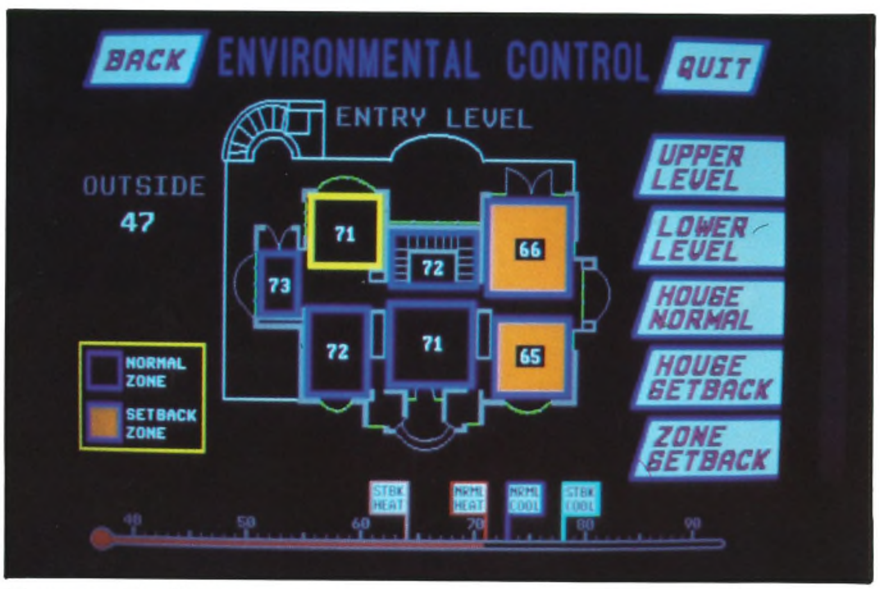

Figure 5. In this home automation system, a floor plan and a thermometer are used to set temperatures in various sections of the house. (Copyright © 1988, 1989, 1990 Custom Command System. Reprinted by permission. All rights reserved.)

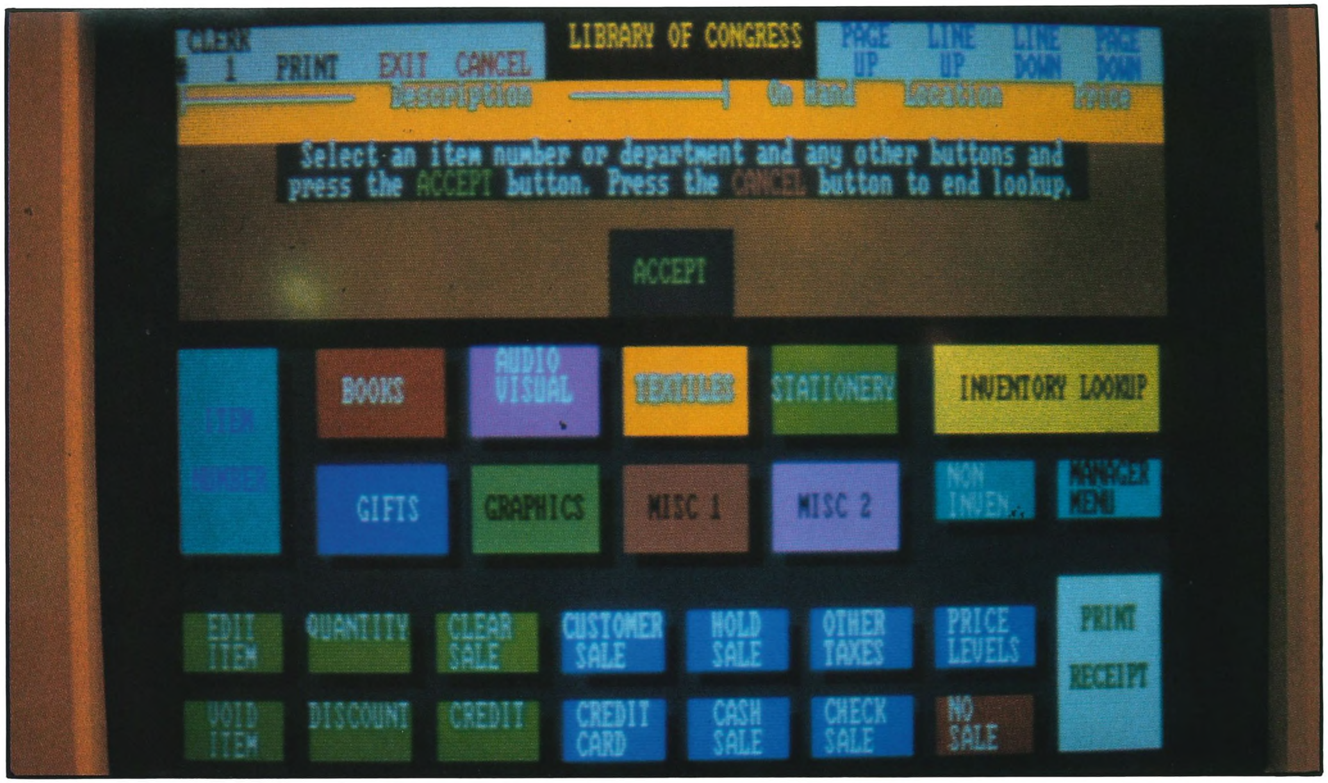

Figure 6. With this cash register the operator specifies products by entering product numbers or using a menu system. Dollar amounts can be entered on a touchscreen keypad. (Copyright © 1988, 1989, 1990 Touch Industries. Reprinted by permission. All rights reserved.)

Figure 7. An example of a process control system: operators use touch-screens to monitor processes and modify parameters if needed.

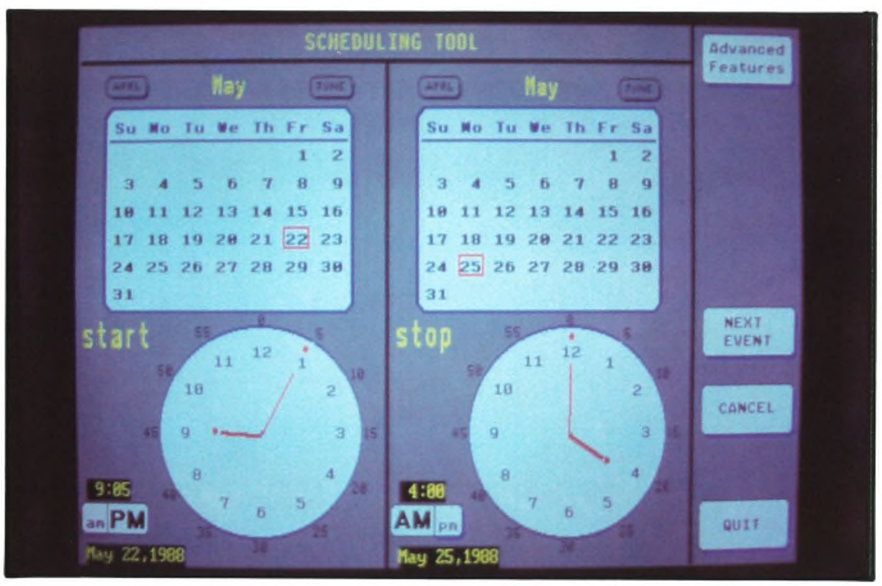

Figure 8. This scheduler allows a device to be turned on and off by selecting days and times. The ON time is set using the left dial, the OFF time using the right. Days can be selected by touching the calendar. Users touch the hands of the clock and rotate them to set the desired time. An AM/PM toggle is provided. (Copyright © 1988 University of Maryland. Reprinted by permission. All rights reserved.)

Figure 9. **PlayPen II, developed in the Human Computer Interaction Laboratory at the University of Maryland, allows users to paint using various patterns and colors using their fingers (sound is also used). (Copyright © 1990 University of Maryland. Reprinted by permission. All rights reserved.)**

Arm fatigue is an important consideration when designing a touchscreen workstation. If the touchscreen is to be used frequently or for extended periods of time, the touchscreen must be mounted at an angle that will reduce fatigue. Screens are typically mounted so that the screen is approximately perpendicular to the surface of the desk. Mounting the screen at an angle closer to horizontal appears to significantly reduce arm fatigue (Sears, 1990; Ahlström, Lenman, & Marmolin, 1991). The best angle has not yet been determined, but mounting the monitor at 30 or 45 degrees from the horizontal appears to help. In addition to reducing arm fatigue, eye strain may also be reduced if the screen is mounted at this angle (Grant, 1987). Providing an elbow rest may also significantly reduce arm fatigue (Ahlström, Lenman, & Marmolin, 1991).

Mounting the screen at different angles results in biases when the users attempt to touch the screen. Users consistently touch below the target when the screen is tilted away from them (Sears, 1990; Beringer & Peterson, 1985). The closer to horizontal that the screen is mounted, the farther below the target users will touch. These studies have indicated that there is also a horizontal bias. One study indicated that these biases may be overcome if the targets are large enough by adjusting the touch locations in the software (Sears, 1990). It is important to realize that these biases will not only depend on the position of the monitor, but also the touchscreen technology, manufacturer, and possibly even the application. These biases are relatively small, and may not have an affect on user performance in most applications, however it may be necessary to test for biases with each new situation.

There are many other considerations when designing a touchscreen workstation. If the monitor is to be mounted inside a protective case, the screen must be mounted to allow users to easily touch the edges of the screen. If the monitor is recessed too far back, users with long fingernails may have difficulty touching the edges of the screen. In the case of a museum application, the designer may consider adding a supplementary display mounted above the touchscreen to allow additional patrons to observe the screen. Other considerations include the height of the desk and chair.

FUTURE DIRECTIONS FOR TOUCHSCREEN RESEARCH AND APPLICATIONS

The future of touchscreen research and applications is exciting. Potential psychology and computer science research topics include: fatigue from extended use, multitouch touchscreens, touchscreen applications for expert users, and improving touchscreen interfaces for data entry.

Research Directions

- *Simultaneous touches:* We have two hands and ten fingers! Just as we use the shift or control keys on a keyboard, simple applications can be envisioned where the users' second hand would be used. A painting program could allow one hand to be painting while the other hand (or even another person) has control of the color. One hand can be selecting the Delete button while the other points at the object or file to be deleted. We can also think of naturally using two fingers to mark a range on a line, create links between objects, mix colors, set attributes, and many other activities. A multitouch touchscreen would also allow overlapping touches to be distinguished, which may significantly improve applications requiring rapid data entry. Multitouch touchscreens are not available commercially but have been explored briefly in research settings. The design of the workstation will be more challenging if both hands must have easy access to the screen. Surface acoustic wave touchscreens may provide the ability to detect at least two simultaneous touches in the near future.
- *Fatigue:* Extended use of touchscreens is fatiguing when monitors are mounted at traditional angles. Current research is exploring the use of arm rests and varying the angle at which monitors are mounted to reduce fatigue. Both of these possibilities are promising.
- *Improved data entry:* Although touchscreens have never been thought of as data entry devices there are many applications where using the touchscreen for data entry may prove advantageous. One of the main advantages of touchscreen interfaces is the flexibility for designers and users. Touchscreen interfaces can be customized for each special value that must be entered offering the user the most appropriate interface.

Touchscreens may never be as fast as a keyboard for typing text but there are many situations where it may be useful to use a touchscreen to enter short strings, such as applications that require limited text entry, and portable computers having a touch-only interface. Preliminary research, discussed above, indicates that users are able to type approximately 25 words per minute using a touchscreen (compared to 58 wpm on a keyboard for the same users) (Sears, 1990). Research must be performed to improve keyboard layout as well as key size, shape, and spacing (Sears, 1990; Sears, Kochavy, & Shneiderman, 1990).

Data entry is not limited to text; it can include telephone numbers, colors, compass directions, etc. Touchscreens may prove to be particularly useful for entering values such as colors and compass directions by allowing users to simply touch the desired color on a color wheel or point on a compass presented on the screen. Instead of forcing users to type telephone numbers on the QWERTY keyboard, a touch-telephone pad could be presented. There are many other values that may be entered more easily using a touchscreen interface than a keyboard or mouse.

- *Using multiple input devices:* There are many situations where using multiple input devices is either convenient or necessary. The most noteworthy being text processing applications. Keyboards are by far the fastest method for entering text, but are slow for cursor positioning. In this situation, using both a keyboard and a touchscreen may prove to be the best way to accomplish the task (similar to how a mouse is currently used). In these situations it is important that the transition from one device to the other as smooth as possible, not only to speed performance but reduce error rates and user frustration. Some research has been conducted to compare using a keyboard with either a mouse or a touchscreen. This research indicates that the touchscreen allows an easier transition than the mouse (Karat, McDonald, & Anderson, 1986).
- *The Tap-Click:* Throughout this chapter landing on and lifting off of the screen have been significant points in the input process. The tap-click offers an additional point in the interaction that may be of use. A tap-click refers to the user touching the screen with a second finger for a brief period of time. This allows users to keep the pointing finger on the screen for future positioning tasks, much like the way a mouse cursor remains on the screen even after a selection.

Two important considerations when using tap-clicks are recognizing a tap-click and knowing where the tap-click occurred. A tap-click occurs when the cursor moves past a minimum distance from where the user was touching and returns to the original touch location in a relatively short period of time. This involves three parameters, the distance the touch location must change by, how close to the point of origin the touch location must return to, and the time allowed for this movement. Since the cursor is moving it may be difficult to recognize where the tap-click was supposed to be located. It is possible that tap-clicks may prove too complicated to use in many situations. Research is needed to perfect the recognition of tap-clicks, the identification of where the tap-click occurred and to measure user acceptance of tap-clicks. Our initial implementation is encouraging. Tap-clicks may be useful when users will be specifying several locations on the screen and do not wish to remove their fingers.

- *Using the Z axis:* Some touchscreens, surface acoustic wave in particular, offer a third dimension. The use of the Z axis has not been explored thoroughly. The Z axis could be used to select the size of an object or text, to act as confirmation, or for many other interesting possibilities. Although the current resolution of the Z axis is very low (16 levels are available, however only three or four are useful), this will surely change as touchscreen manufacturers refine their products.

- *Touchscreen hardware:* Manufacturers are working on many improvements including reduced parallax, higher resolution, and increasing the amount of light transmitted through the touchscreen. This research will be necessary as the resolution of displays and the range of touchscreen applications increase. Research will continue on these traditional problems as touchscreens become more popular. In addition, the touchscreen industry is creating new touch input devices including touch sensitive remote controls with programmable displays and the UnMouse. The UnMouse, by MicroTouch, is a touch sensitive pad that can be used in place of a mouse to move the cursor. It also has features not available on a mouse including the ability to act as a graphics tablet and as additional special function keys. Touchscreen manufacturers will also play an important roll in the development of multitouch and three-dimensional touchscreens.

Touchscreen Applications

The future of touchscreen applications is promising. When touchscreens were first introduced, they were considered novel input devices that were only suitable for selecting relatively large areas of the screen. Touchscreens are now being used in many applications including cash registers, automobile controls, and home automation systems. Touchscreens are also being used in hypertext systems and strictly for fun in games.

The number of applications for touchscreens has increased, and designers are moving beyond simple button pushing tasks. Applications are taking advantage of the touchscreens' ability to present novel interfaces for traditional tasks. One example of such an interface is a device scheduler developed by the Human-Computer Interaction Laboratory and Custom Command Systems. This interface presents a time line and allows users to schedule events by placing start and stop markers (Figure 4) (Plaisant & Shneiderman, 1989). Another example is a finger painting program also developed at the Human-Computer Interaction Laboratory that allows users to draw using various colors and shapes (Figure 9).

Recent studies that indicate that touchscreens can be used for high-resolution tasks may lead to many new applications. Dragging objects on the screen and using freehand movements as input will also allow touchscreens to be used for many new applications. The increased availability of touchscreen toolkits and development packages will lead to faster development of touchscreen interfaces. User Interface Management Systems (UIMS) will be developed to take increased advantage of touchscreens, further stimulating touchscreen use. Refined theories and better taxonomies of tasks may help guide designers toward novel implementations. In short, the future of touchscreens looks bright and feels good.

ADDITIONAL SOURCES OF INFORMATION ABOUT TOUCHSCREEN RESEARCH

We have referenced a small subset of the research that is relevant to working with touchscreens. See the following for additional information: Arnaut & Greenstein, 1986; Baggen & Snyder, 1988; Battenburg, 1989; Beringer, 1989, 1990; Beringer & Bowman, 1989; Bolger, 1989; Buxton, Will, & Rowley, 1985; Ellis, Huang, & Buzzard, 1986; Gaertner & Holzhausen, 1980; Gould, Greene, Boies, Meluson, & Rasamny, 1990; Grant, 1987; Greenstein & Arnaut, 1988; Levine & Garwin, 1983; Mahach, 1989; Shneiderman, Brethauer, Plaisant, & Potter, 1989; Schulze & Snyder, 1983; Valk, 1985; Weiman, Beaton, Knox, & Glasser, 1985; Weisner, 1988; Whitfield, Ball, & Bird, 1983; Wolf, 1988; and Wolf & Marrel-Samuels, 1987. This is only a partial list of relevant research, but it should provide a useful starting point to build on.

ACKNOWLEDGMENTS

We would like to thank Arthur Carroll for providing valuable information about the various touchscreen technologies, the reviewers and editors for their thoughtful comments, and John Smelcer and Leslie Carter for reviewing the final draft. We appreciate the partial support of research by NCR Corporation, Museum of Jewish Heritage, and the Maryland Industrial Partnerships program.

REFERENCES (ALPHABETICAL WITH SUBJECT INDICATORS)

C - Comparisons of various input devices
G - General touchscreen papers
H - Touchscreen hardware
R - Review articles
S - Selection Strategies
T - Touchscreen only studies
W - Workstation

C Ahlström, B., & Lenman, S. (1987). Touch screen, cursor keys, and mouse interaction. In *Work with display units* (pp. 831–837). New York: Elsevier.

W Ahlström, B., Lenman, S., & Marmolin, T. (1991). Overcoming touchscreen user fatigue by workplace design. IPLab Interaction and Presentation Laboratory, Department of Computing Science, Royal Institute of Technology, S-100 44 Stockholm, Sweden.

Apple Computer Inc. (1987). *Human Interface Guidelines: The Apple Desktop Interface*. Reading, MA: Addison-Wesley.

T Arnaut, L., & Greenstein, J. (1986). Optimizing the touch tablet: The effects of control-display gain and method of cursor control. *Human Factors, 28*(6), 717–726.

G Baggen, E., & Snyder. H. (1988). A human factors evaluation of current touch-entry technologies. *Society for Information Display 88 Digest*, pp. 259–262.

C Battenberg, J. (1989). Touch screen versus keyboard: A comparison of task performance of young children. *Journal of Special Education Technology*, X(1), 24–28.

WT Beringer, D., & Peterson, J. (1985). Underlying behavioral parameters of the operation of touch-input devices: Biases, models, and feedback. *Human Factors, 27*(4), 445–458.

W Beringer, D., & Bowman, M. (1989). Operator behavioral biases using high-resolution touch input devices. *Proceedings of the 33rd Annual Meeting of the Human Factors Society*, pp. 320–322. Santa Monica, CA.

CS Beringer, D. (1989a). Touch panel sampling strategies and keypad performance comparisons. *Proceedings of the 33rd Annual Meeting of the Human Factors Society*, pp. 71–75. Santa Monica, CA.

C Beringer, D. (1989b). *Parameters influencing the airborne application of touch devices for data entry and retrieval* (Tech. Rep. HFRL-89-1/MDHC-89-1). Las Cruces: New Mexico State University, Human Factors Research Lab.

T Beringer, D. (1990, April 18–20). Target size and location effects on touch panel performance: limits on interpretation and inference. *Proceedings of the Twelfth Symposium Psychology in the Department of Defense* (pp. 122–126). Colorado Springs, CO: US Air Force Academy.

G Bolger, J. (1989, July). CD-ROM delivery: Breaking new ground. *Optical Information Systems*, pp. 214–216.

CT Buxton, W., Will, R., & Rowley, P. (1985). Issues and techniques in touch-sensitive tablet input. *Proceedings of SIGGRAPH '85* (pp. 215–224). New York: ACM.

H Carroll, A. (1986, November). Touch technology: Variety of systems spur maturity. *Society for Information Display*, pp. 15–20.

HG Carroll Touch. (1989). *Touch Handbook*. Round Rock, TX: Carroll Touch Inc.

C Ellis, L., Huang, P., & Buzzard, I. M. (1986). Touchscreen versus keyboard for menu based food coding. In B. Blum & M. Jorgensen (Eds.), *MEDINFO'86* (pp. 999–1003). New York: Elsevier Science Publishers.

Fitts, P. M. (1954). The information capacity of the human motor system in controlling the amplitude of movement. *Journal of Experimental Psychology, 70,* 193–242.

T Gaertner, K. P., & Holzhausen, K. P. (1980). Controlling air traffic with a touch sensitive screen. *Applied Ergonomics, 11,* 17–22.

H Garwin, R. L., & Levine, J. L. (1989). Light-sensitive pen to substitute for finger in laser-scanned touch screen. *IBM Technical Disclosure Bulletin, 32*(3B), 435–436.

T Gould, J., Greene, S., Boies, S., Meluson, A., & Rasamny, M. (1990). Using a touchscreen for simple tasks. *Interacting with Computers, 2*(1), 59–74.

W Grant, A. (1987). The computer user syndrome. *Journal of the American Optometric Association, 58*(11), 892–901.

R Greenstein, J., & Arnaut, L. (1988). Input devices. In Helander (Ed.), *Handbook of human-computer interaction* (pp. 495–519). New York: Elsevier.

WT Hall, A., Cunningham, J., Roache, R., & Cox, J. (1988). Factors affecting performance using touch-entry systems: Tactual recognition fields and system accuracy. *Journal of Applied Psychology, 73*(4), 711–720.

C Karat, J., McDonald, J., & Anderson, M. (1986). A comparison of selection techniques: Touch panel, mouse, keyboard. *International Journal of Man-Machine Studies, 1,* 73–92.

HS Levine, J. L., & Garwin, R. L. (1983). *A laser-scanned touch input device.* For Display '83 in Japan.

R Mahach, K. (1989). *A comparison of computer input devices.* Masters thesis, George Mason University, Fairfax, VA.

C Muratore, D. A. (1987). Human performance aspects of cursor control devices. *Mitre Corporation Working Paper 6321.* Houston, TX.

S Murphy, R. (1987, November). *Evaluation of methods of touch screen implementation for interactive computer displays.* Abstract presented at 2nd International Conference of Human-Computer Interaction, Honolulu, HI.

Nielsen, J., & Lyngbaek, U. (1990). Two field studies of hypermedia usability. In C. Green & R. McAleese (Eds.), *Hypertext: Theory into practice II.* Norwood, NJ: Ablex.

Norman, D. (1988). *The Psychology of Everyday Things.* New York: Basic Books.

Norman, D., & Fisher, D. (1982). Why alphabetic keyboards are not easy to use: Keyboard layout doesn't much matter. *Human Factors, 24,* 509–515.

C Ostroff, D., & Shneiderman, B. (1988). Selection devices for user of an electronic encyclopedia: An empirical comparison of four possibilities. *Information Processing & Management, 24*(6), 665–680.

HR Pickering, J. A. (1986). Touch-sensitive screens: The technologies and their applications. *International Journal of Man-Machine Studies, 25,* 249–269.

T Plaisant C., & Shneiderman, B. (1989). *Scheduling ON-OFF home control devices: Design issues and usability evaluation of four* *touchscreen interfaces* (CAR-TR-472, CS-TR-2352). College Park, MD: University of Maryland. (To appear in: *International Journal of Man-Machine Studies*)

T Plaisant, C. & Sears, A. (1991). *Touchscreen interfaces for flexible alphanumeric data entry* (Technical Report CAR-TR-585, CS-TR-2764). College Park, MD: University of Maryland.

S Potter, R., Berman, M., & Shneiderman, B. (1989). An experimental evaluation of three touch screen strategies within a Hyperties database. *International Journal of Human-Computer Interaction, 1*(1), 41–52.

S Potter, R., Weldon, L., & Shneiderman, B. (1988). Improving the accuracy of touch screens: an experimental evaluation of three strategies. *Proceedings of the Conference on Human Factors in Computing Systems* (pp. 27–32), New York: ACM SIGCHI.

TG Sears, A., Kochavy, Y., & Shneiderman, B. (1990). Touchscreen field specification for public access database queries: Let your fingers do the walking. *Proceedings of the ACM Computer Science Conference '90* (pp. 1–7). New York: ACM.

T Sears, A. (1990). *Revising touchscreen keyboards: Design issues and a comparison with other devices* (CAR-TR-515, CS-TR-2536). College Park, MD: University of Maryland. (To appear in: *Interacting with Computers*)

C Sears, A., & Shneiderman, B. (1991). High-precision touchscreens: Design strategies and comparison with a mouse. *International Journal of Man-Machine Studies, 34*, 4, 593–613.

T Sears, A., Revis, D., Swatski, J., Crittenden, R., & Shneiderman, B. (1991). *Investigating touchscreen typing: The effect of keyboard size on typing speed* (Technical Report CAR-TR-553, CS-TR-2662). College Park, MD: University of Maryland.

R Sherr, S. (1988). *Input devices*. San Diego, CA: Academic Press.

G Shneiderman, B. (1987). *Designing the user interface: Strategies for effective human-computer interaction*. Reading, MA: Addison-Wesley.

G Shneiderman, B., Brethauer, D., Plaisant, C., & Potter, R. (1989, May). The Hyperties electronic encyclopedia: An evaluation based on three museum installations. *Journal of the American Society for Information Science, 40*(3), 172–182.

C Schulze, L., & Snyder, H. (1983). *A comparative evaluation of five touch entry devices* (Tech. Rep. HFL-83-6). Blacksburg, VA: Virginia Polytechnic Institute and State University, Department of Industrial Engineering and Operations Research.

H Stone, M. D. (1987, August). Touch-screens for intuitive input. *PC Magazine*, pp. 183–192.

T Valk, M. A. (1985). An experiment to study touchscreen button design. *Proceedings of the Human Factors Society 29th Annual Meeting* (pp. 127–131). Santa Monica, CA.

T Weiman, N., Beaton, R., Knox, S., & Glasser, P. (1985). *Effects of key layout, visual feedback, and encoding algorithm on menu selection with LED based touch panels* (Tech. Rep. HFL-604-02). Beaverton, OR: Human Factors Research Lab, Tektronix Laboratories.

G Weisner, S. (1988). A touch-only interface for a medical monitor. *Proceedings of the Human Factors Society—32nd Annual Meeting* (pp. 435–439). Santa Monica, CA.

C Whitfield, D., Ball, R. G., & Bird, J. M. (1983). Some comparisons of on-display and off-display touch input devices for interaction with computer generated displays. *Ergonomics, 26*(11), 1033–1053.

C Wolf, C. (1988). A comparative evaluation of gestural and conventional interfaces. *Proceedings of the Human Factors Society 22nd Annual Meeting* (pp. 272–277).

C Wolf, C., & Marrel-Samuels, P. (1987). The use of hand-drawn gestures for text editing. *International Journal of Man-Machine Studies, 27,* 91–102.

APPENDIX A—TOUCHSCREEN MANUFACTURERS (PARTIAL LIST)

W. H. Brady Co.
8225 W. Parkland Ct.
P.O. Box 571
Milwaukee, WI 53201
Telephone: (414) 355-8300

Carroll Touch
P.O. Box 1309
Round Rock, TX 78680
Telephone: (512) 244-3500

Ellinor Technology
Arkwright Road
Reading, Berks.
United Kingdom RG2 0EA
Telephone: 011-44-734-311-066

Elographics, Inc.
105 Randolph Road
Oak Ridge, TN 37830
Telephone: (615) 482-4100

IBM
Contact local distributor

John Fluke Manufacturing Co.
P.O. Box 9090
Everett, WA 98206
Telephone: (206) 347-6100

MicroTouch Systems Inc.
55 Jonspin Road
Wilmington, MA 01887
Telephone: (508) 694-9980

TSD Display Products
35 Orville Drive
Bohemia, NY 11716
Telephone: (516) 589-6800